

DEPARTMENT: VISUAL COMPUTING: ORIGINS

Computer Graphics at Evans & Sutherland and Pixar

Mickey W. Mantle , Napa Valley, CA, 94558, USA

From the Cradle of Computer Graphics at the University of Utah to Pixar Animation Studios, my 50+ year professional career has been a parabola of many groundbreaking projects, as well as startup companies—some that have become household names, including Brøderbund Software, Gracenote/SONY, and others. It has been an exciting and rewarding journey not only because of the bleeding edge technology and innovative products I worked on, but also due to the privilege of working with, managing, and leading people in these companies. I reflect on part of this journey in this article and my book “Managing The Unmanageable”.

In 2023, I attended IEEE Milestone events that commemorated the following:

- › The University of Utah (U of U) Computer Science (CS) Department for the Development of Computer Graphics and Visualization Techniques 1965–1978.^a
- › Pixar for Creating RenderMan: Pixar’s Impact on Computer Graphics.^b

I also attended ACM SIGGRAPH 2023: Celebrating 50 Years of SIGGRAPH Conferences.¹

Attending all of these events in one year reminded me how important the U of U CS Department, Evans & Sutherland (E&S), and Pixar were as stops on my career and how important SIGGRAPH was to my personal and professional development.

This article tells my story through the stories of the bleeding edge 2-D/3-D computer graphics projects and

products that I worked on at E&S and Pixar. I worked with a great many people—and many great people—and I apologize up front for leaving out so many of them who worked with me and contributed to these projects. Article length constraints limit me from mentioning the many who contributed in important ways.

EDUCATION AND EARLY CAREER (1971–1973)

I was hired at E&S in the fall of 1973 as a member of the Picture System team. E&S was interested in me because I had some fairly unique skills for that time—a background in 3-D computer graphics and extensive real-time programming experience.

My background in 3-D graphics came from the U of U CS computer graphics courses I took from Ivan Sutherland in 1970–1971. Ivan’s classes consisted of fewer than 10 grad students, some of whose work would be a primary driver of computer graphics innovation for decades to come: Ed Catmull (Pixar), Henry Fuchs (UNC), Fred Parke (University of Texas), and Russ Athay (E&S), among others. Henri Gouraud (of Gouraud shading fame) was one of the teaching assistants for the course. At the time, I did not realize I was standing on the precipice of a sea-change of what would shape the next 40+ years of graphics technology.

My programming experience to that point came from the education I got at the U of U, which gave me

^a[Online]. Available: https://ethw.org/Milestones:Development_of_Computer_Graphics_and_Visualization_Techniques,_1965-1978

^b[Online]. Available: <https://renderman.pixar.com/news/renderman-receives-ieee-milestone-award>

a great foundation in programming, and from my first job out of college. After graduating in 1971, I was hired by Kenway Engineering as one of a team of electrical/mechanical engineers and programmers tasked with creating an automated tracking and delivery system for a six-acre Computer-Assisted Aircraft Accessories Rework Facility located on North Island Naval Base, Coronado, CA.

This system, implemented in Assembly Language and FORTRAN, controlled four stacker cranes plus a six-acre overhead conveyor system with 20 delivery elevators, one for each repair station. The system maintained an inventory of parts and repair kits that were loaded onto special pallets that were checked into/retrieved from the four-story storage system by one of the stacker cranes from/to the conveyor system automatically upon demand from a repair station.

Kenway was my first “real programming job,” where I honed real-time programming skills writing interrupt device drivers and “reentrant” FORTRAN code on IBM 1830 clone computers (General Automation 18/30s) that were configured as fully redundant computers (mirrored hard disk storage systems with real-time data synchronization and automatic failover if the primary system failed).

By the time this system was installed in 1972, I had risen to be the programming lead and oversaw the installation and support of this six-acre system. I was extremely lucky to have had this as my first programming job. I had learned real-time programming overcoming error-prone unreliable hardware, plus technical leadership skills.

E&S, 3-D GRAPHICS (1973–1984)

I joined E&S in 1973 just as the company began to split into two divisions: Interactive Systems Division (Picture Systems) and the Simulation Systems Division (Flight Simulators). The technologies in the two divisions were quite different. The Picture Systems were end-user, vector devices until the early 1980s. They were used primarily for creating and editing (e.g., computer-aided design). The simulators turned to raster displays in the mid-1970s. Users required more realistic displays representing runways, buildings, etc.

While working at E&S, I worked primarily on Picture System products, beginning as a systems programmer and then moving through a series of product related jobs and special projects. When I left the company, I was the Interactive Systems Software Products Manager. Five generations of Picture System products were produced during this period, as shown in Figure 1. These products introduced a generation of companies, academics,

researchers, and students to the power of real-time, interactive 3-D graphics.

Picture System (1976–1984)

During much of my time at E&S, I was paired with Steve McAllister (an incredibly great programmer and person). He and I worked closely to develop the graphics software package (GSP) for the then new E&S 3-D graphics machine—the Picture System. The GSP let a programmer specify the 3-D content and its 3-D view, and how user input should be gathered. It was intended to be an interactive 3-D device for industry, government, and education.

Steve and I worked feverishly to complete the GSP library for RT-11 (the OS for DEC’s PDP-11 computers) device drivers, and user documentation for the product. Jim Callan (responsible for Chapter 2) and I undertook the task of writing the *Picture System User’s Manual*. The Manual introduced users to the concepts of 3-D graphics and the specific procedure calls needed to create and interact with 2-D/3-D line drawing programs on the Picture System hardware. In those days before MS Word and Adobe products, I had to use my TECO and RUNOFF skills (honed at the U of U) to create “professional quality” drafts of the document for review and printing.

Picture System 2 (PS2) (1976–1983)

The Picture System was a landmark system, but it was costly to produce and challenging to support.

E&S decided to create a second generation of the product, the PS2, with newer technology and a more integrated design. The display and hardware chassis had a modern industrial design esthetic and were relatively easy to produce. This second generation system also had a new calligraphic (vector) display that was brighter and easier to maintain.

The PS2 system was compatible with the earlier system at the source code level, so existing software could run on this next generation hardware without major modification. This made it easy for existing customers to transition to the new system. However, the PS2 still required a PDP-11 to host the system and application software.

Although the GSP library procedure calls were source code compatible, under the covers the software and hardware were quite different. Steve McAllister and I spent a lot of time creating the PS2 version of GSP so that it masked the hardware differences from the customer programmers. I also updated the PDP-11 RT-11 device drivers and wrote RSX-11M and VAX/VMS device drivers for the PS2.

We created a new *PS2 GSP User’s Manual*, updating the original documentation as required and, as before, created print-ready documents using TECO and RUNOFF.



FIGURE 1. Five generations of Picture System 3-D graphics systems.

In addition, I created the *PS2/PDP-11 Reference Manual* detailing the low-level hardware interfaces of the PDP-11 interface card, as well as the programmable hardware of the PS2 that was used by the GSP. This enabled other software interfaces to be developed for the PS2, such as the Unix GSP created by Tom Ferrin at the UC San Francisco Department of Pharmacology.

One other thing Steve McAllister and I did that was important for supporting the PS2 was to create a set of PS2 diagnostic programs that tested whether the hardware was working properly. This also included a set of display calibration programs.

At that time, RT-11 was a popular single-user OS for the PDP-11, and the source code for RT-11 was included in the operating systems distribution package. So (without checking with anyone at DEC or E&S), I decided to minimally alter the RT-11 OS source code so that when it was booted up it came up with a different input prompt and identified itself as the “E&S Diagnostic System.” We packaged this slightly modified RT-11 and all of the PS2 diagnostic programs on bootable media, and for many years, it was distributed to all E&S customer support staff who maintained PDP-11/PS2 systems.

Multipicture System (MPS) (1977–1979)

In an effort to further reduce the “per-seat cost” for a system, E&S decided to create the MPS, which by adding a second display and set of input devices would allow two users to share the computational components of a single PS2. This new product allowed companies, such as McDonnell-Douglas, Boeing, and others (including non-aerospace companies) to install MPS systems at about half the cost of a PS2 system on a per-seat basis. It further reduced cost by allowing one or more MPS systems (each with two users) to be hosted by a single PDP-11 or VAX-11 minicomputer.

The MPS required a new software device driver to manage the shared MPS resources on a PDP-11/70 class machine running RSX-11M. But many customers (and specifically McDonnell-Douglas) wanted to run the MPS systems from VAX-11/780 class minicomputers running VAX-11/VMS.

Gary Watkins, my then boss at E&S, was primarily responsible for developing the MPS system. As the MPS was a significant new development project, he decided to develop the VAX/VMS device driver in parallel with the PDP-11/RSX-11M version and called upon the E&S Mountain View Office of software specialists to develop the device drivers.

John Warnock, one of Dave Evan’s Ph.D. students from the U of U, established the Mountain View, CA E&S Office to develop specialized software solutions for various E&S projects. John recruited a small team of excellent programmers and problem solvers [most of who had previously worked for him developing the

operating system for the Illiac IV,^c the first single instruction, multiple data (SIMD) computer: John Gaffney, PJ Zima, and Kristy Barton.

This team worked closely with the Simulation Division of E&S, which developed real-time visual simulation systems for commercial companies and government agencies. This group was instrumental in developing tools (most notably the Design System^d) to efficiently create 3-D models for the E&S NASA Space Shuttle manipulator arm simulator, and for the Computer Aided Operations Research Facility (CAORF) New York Harbor simulator to train harbor pilots on the simulated bridge of a ship (with five-screen real-time shaded images—one for each bridge window) as it navigated through the harbor.

PJ Zima agreed to take on the task of developing the PDP-11/RSX-11M and VAX-11/VMS MPS device drivers. He patterned them after the existing RSX-11M and VMS/PS2 device drivers I had written, but enhanced them to support multiple users on the newly configured MPS. This required no new hardware design, only configuring a PS2 with multiple displays and user input devices.

DARPA Project (1977–1979)

While PJ Zima set out to design and implement the RSX-11M and VMS MPS Device Drivers, I was asked to relocate to California to work with John Warnock on a special project that E&S had received from DARPA, and to consult and help oversee PJ’s work on the MPS device drivers he was writing.

The DARPA project was to develop a demonstration of an unpublished visible surface algorithm by Sutherland.² John Warnock and I implemented Ivan’s object space visible-surface algorithm with real-time presentation of the object space models on a PS2 and shaded renderings (on the raster display of an E&S video frame buffer) of the models in a user-selected view on the PS2 display. We used the CAORF New York Harbor model as the visual test database and the Design System to create a database compiler to compile other 3-D scenes and models.

Not long after PJ finished developing the MPS device drivers, John announced he was leaving E&S to join Xerox PARC. When he left, the California E&S office closed. John Gaffney and PJ left E&S, remained in California, and later joined Adobe Systems after it was founded.

^c[Online]. Available: https://en.wikipedia.org/wiki/ILLIAC_IV

^dThe Design System was an extremely versatile programming system based upon a reverse polish token interpreter, which was instrumental in Warnock’s design and implementation of Postscript years later. One of the first things Warnock did upon founding Adobe Systems was to license IP of the Design System from E&S.

I moved back to Salt Lake City to finish up the DARPA project (with some critical help from Gary Watkins) and rejoined the software team at E&S.

Picture System 300 (PS300) (1980–1984)

Although the MPS was successful, the Interactive Systems division of E&S (i.e., the Picture System group) became obsessed with finding ways to make more “cost effective” solutions for our large customers (i.e., McDonnell-Douglas, Boeing, etc.).

Carl Ellison, a brilliant systems programmer had worked at the U of U for many years before joining E&S. Carl proposed a system that would enable customers with any type of computer to transfer 3-D object definitions to PS300 memory. The PS300 would then display and locally manipulate the 3-D objects without burdening the host computer.

This introduced a new programming technique, which came to be known as function networks, which required, in a data-driven fashion, transferring the 3-D coordinate data of objects to the PS300, along with instructions for how to interact with and manipulate that data.

On powerup, the PS300 “firmware” (programmed in Pascal) was loaded from a floppy disk internal to the system. Once 3-D data were downloaded to the system, the PS300 was capable of doing 3-D real-time calligraphic display and interaction independent of the slow RS-232 or other serial interfaces (e.g., IBM 3270/3278) between the host and the PS300.

While extremely innovative, the PS300 suffered from the following user satisfaction issues:

- The time to download complex 3-D datasets was unreasonably long!
- The programming paradigm was totally different from that of the earlier PS2 and MPS systems.
- Learning a new interaction methodology (“function networks”) was unique to this system, complex to learn (and debug), and generally not intuitive.

But the PS300 also had strengths including:

- Outstanding industrial design of its display and hardware cabinetry.
- Hardware design optimized for manufacture and field support at scale.
- Varied set of peripheral devices for interactive input (keyboard, buttons, tablet, etc.)
- Superior monochrome and color shadow mask analog line drawing displays.
- A legacy of good will among the E&S customer base.

The PS300 was a very innovative product that fundamentally did not meet the needs of some of E&S

existing customers—namely, to be able to accept large 3-D datasets quickly.

In the face of mounting objections by PS1 and PS2 customers to the slow serial line interface from host computers to the PS300, I championed the creation of a parallel interface board for the PS300 to interface to DEC PDP-11 and VAX-11 Unibus. Other interface boards were created as well, including one that emulated an IBM 3278 terminal, which has a serial protocol but was connected via coax cables making the serial data transfer rate much faster than the original RS-232 interface to the PS300. Fortunately, sales of the PS300 demonstrated that the slow data transfer rate wasn’t a problem for many customers (although significant engineering time was spent developing higher bandwidth interfaces to the system in the years after its initial release).

However, for many PS2 customers, the software incompatibility of the PS300 remained a problem.

PS300 Function Networks (1982)

PS300 customers began to report that they had difficulty with the function network local programming paradigm that was core to the design of the system.

To help improve this situation, I contracted with Alan Davis, then an Associate Professor at the U of U, to write a document that provided a clear introduction to “data driven” programming using the PS300 Function Networks: *An introduction to Data-Driven Programming Methodology For PS 300 Users*, by Alan L. Davis (1982–1983).

This document was distributed to PS300 users and was helpful in introducing them to the concept and application of data-driven programming.

PS320 (1983)

The PS320 was the functional successor to the MPS; it was essentially a PS300 with a second display controller and display and a second set of input devices, allowing the system to support two users at the same time. Since the system was in essence shared, the processing capacity per user was reduced and occasionally some perceptible delays could occur.

However, the result was a lower “seat cost” per user. Many of these systems were sold to automotive and aerospace companies.

This was the last Picture System that was sold with only a calligraphic/vector display. With the advent of low-cost dynamic RAM memories, graphics display systems were rapidly converting to raster technology.

PS340 (1984)

The PS340 introduced a calligraphic color monitor as well as an optional raster monitor on which shaded images could be displayed. The firmware for this

configuration included a “visible surface” algorithm, which allowed 3-D polygonal images to be dynamically moved and interacted with on the calligraphic monitor. When a view was selected for display, the PS340 would compute a shaded image from the data and display it on the raster monitor. This was a product that my DARPA project had functionally prototyped years before.

Since the PS340 had both calligraphic and raster displays, it addressed users’ needs to display and interact with vectors and display high-quality shaded images.

PS330, PS350, PS390 (1985–1988)

E&S released several additional Picture System products that introduced additional display capabilities including antialiased raster line drawing capabilities. These were based upon the PS300 products with alternate display boards (i.e., Shadowfax boards).

REALimage 3-D Chips and Add-in Boards (1987–1988)

E&S introduced 3-D graphics chips for add-in board manufacturers, and 3-D graphics boards that used the REALimage chips for personal computer manufacturers.

Other E&S Products and Projects

E&S made dozens of other simulation systems, software systems, and even a SuperComputer, not discussed here. The E&S products listed above focus on Picture System products and related products made to address the same markets (e.g., automotive, aerospace, academic research, etc.).

During my time at E&S, I was involved in several other projects that were related to our Picture System products. Here is a partial list of them.

Picture System Lorgnette for Stereo Viewing (1974)

Three-dimensional molecular modeling needed stereo viewing and/or color to better investigate molecular structures, and the Lorgnette was a way to provide this capability. A Lorgnette is a black and white or color stereoscopic viewer using a rotating segmented disc. To produce stereo images, the Picture System synchronized two offset images (left-eye and right-eye) on the screen with the Lorgnette, which is held in front of the face and alternately masks each eye 30 times per second, as shown in Figure 2.

To produce color images, the Picture System synchronized three coincident images of varying intensity that represented the R, G, and B color components of the image synchronized with the rotating Lorgnette RGB disc. The user looked directly at the Lorgnette, which was held so that the R, G, or B segments were in front of the eyes when each corresponding color image appeared. The Lorgnette was developed for E&S’ first products, the Line Drawing Systems (LDS-1



FIGURE 2. E&S Lorgnette Stereo Viewer.

and LDS-2). The rotating disks were reportedly “manufactured” by Ivan in a frying pan in his kitchen.

I wrote the device driver, software package, and manual that supported the Lorgnette on the Picture System.

Teaching Picture System Courses (1974–1976)

Steve McAllister and I put together a PS2 training course and presented it to customers at E&S headquarters, as well as throughout the USA and other parts of the world.

Video Frame Buffer (1973–1977)

This device was the first commercial video (raster) frame buffer.^e The arrival of dynamic RAM chips

^e[Online]. Available: https://en.wikipedia.org/wiki/History_of_computer_animation#Towards_3-D:_mid-1970s_into_the_1980s

made digital memory systems for frame buffers and raster display systems possible. The E&S frame buffer had a resolution of 512 by 512 pixels in 8-bit grayscale or color through the use of a color-map. It was designed by Ivan Sutherland, Jim Kajiya, Gary Demos, and Ed Chandle. Making it a product required subsequent engineering by Don Garner and Alan Tanner.

Later, NYIT created the first full-color 24-bit RGB framebuffer by using three of the E&S frame buffers linked together as one device connected to a PDP-11. They also created a 1024x1024 8-bit frame buffer by using four of the E&S frame buffers (one for each quarter of the 1024x1024 image grey scale or color through the use of a color map) linked together as one unit connected to a PDP-11.

An E&S video frame buffer was integral to my DARPA Project.

3-D Computer Graphics Standards (1979–1983)

I deemed it important to keep an eye on standards efforts and regularly attended the Computer Graphics Standards meetings held at SIGGRAPH starting in the late 1970s. I was a skeptic about committee-based Computer Graphics Standards (e.g., SIGGRAPH Core, PHIGS). This opinion was later reinforced by the subsequent success of defacto Graphics Standards (and implementations), such as GL/OpenGL, PostScript, and RenderMan.

Digistar (1976–Today)

After the PS2 was released, Steve McAllister and I visited our local planetarium. I bought a Star Catalog that included the position and brightness of thousands of the brightest stars observable from Earth. I gave the book to Steve, and he entered the data in the book (right ascension, declination, and star type) into a file and wrote a program that would convert it into data suitable for display on the PS2 (x , y , z , and intensity, inferring the z coordinate from the type of star and its intensity).

Steve's Star Catalogue program showed the possibility of using the PS2 as a way to view a real star field—a digital planetarium. Thus, began a decades long journey for Steve and Brent Watson to make Digistar, the first digital planetarium.

Their journey was not an easy one; it had only sporadic company support, but Steve and Brent persevered, and finally E&S turned Digistar into a product. E&S acquired Spitz (a dome maker) and created the first digital planetarium.

As years went on, E&S sold off their Simulation Systems division to Rockwell (2006) and closed

down all other products. E&S, a company that had had a broad product portfolio, became a digital planetarium company, with dozens of Digistar planetariums worldwide. In 2022, E&S was acquired by COSM and is now creating incredible LED domes for Planetariums as well as immersive 3-D experience centers around the world.

PS3 Proposal (1983)

I wrote a proposal to recast the PS300 hardware as the functional successor to the PS1 and PS2—the PS3. Here is the first paragraph of the PS3 proposal.

PS3 PROPOSAL

A PS3 is the functional successor to the PS1/PS2 products. It is closely coupled to the host computer with little or no software provided to make it either easily used or user friendly. A PS3 is configured from existing PS300 components and is maintained in the same manner as a PS300. A PS3 could be considered to be unbundled PS300 hardware, although it is not recommended that unbundling of the PS300 hardware/software be done. A PS3 is rather just an alternate mode that a single-user PS300 could be used in.

Unfortunately, I never received a clear response to my proposal. But it wasn't approved.

PS300 Unix Port—PS360 (1983)

I embarked on a Skunk Works^f project to port Unix to the PS300 controller board. Since this PS300 board was essentially a self-contained Motorola 68010 computer system, I commissioned a port of Berkeley Unix by Berkeley Systems, a company in the SF Bay area.

The project was code named PS360.

After the port was completed, I set up a system to demonstrate it on a PS300 system.

A short time after that I received word that E&S was NOT interested in making general purpose computers with graphics capability and the PS360 project was shut down.

Interactive μ graphics Proposal (1984)

I was convinced that 3-D interaction combined with raster graphics was going to be big in the personal

^fSkunkworks projects are low-profile, often secretive, projects to prove the feasibility of a new process or technology. The name is taken from Lockheed's famous Skunkworks in Burbank that created planes, such as the P-38, U2, SR-71 Blackbird, and many others.

computer space, and I thought that E&S could be an important company in 3-D graphics for PCs.

The IBM PC was introduced in August 1981 and by 1983 was selling almost 200,000 units a quarter. So I did whatever research I could and made an "interactive μ graphics" proposal, which would be a collection of graphic processor and video display boards for doing 2-D and 3-D graphics on the IBM PC and other microcomputer systems.

I naively sent my seven page proposal to E&S executives hoping to get some sort of positive response to the idea, if not to this exact proposal. Figure 3 shows the original overview of this proposal.

I finally got a response to my proposal and I clearly remember talking to Dave Evans about it. He explained to me that our sales staff was not capable of selling such a low-priced product that would carry little or no commission. At the time, I didn't know enough about alternate distribution channels to counter his argument. Years later, E&S tried to enter

the personal computer market with limited success, releasing their Freedom Graphics board and later their REALimage 3-D chips and 3-D Graphics boards. (See the "REALimage 3-D Chips and Add-in Boards (1987–1988)" section.)

Time to Leave E&S

My years at E&S were among the best of my career. Being a key contributor to five generations of Picture System products, teaching Picture System training courses in the USA and other countries around the world, working with John Warnock and Gary Watkins on a DARPA project to deliver a new visible surface algorithm based upon Ivan Sutherland's unpublished paper, and hiring a generation of engineers who made significant contributions at E&S was a life-changing experience for me.

Deciding to leave E&S was bittersweet. E&S was more like family than any other company I worked at. My friendships with others at E&S expanded to include shared vacations together, teaching guitar

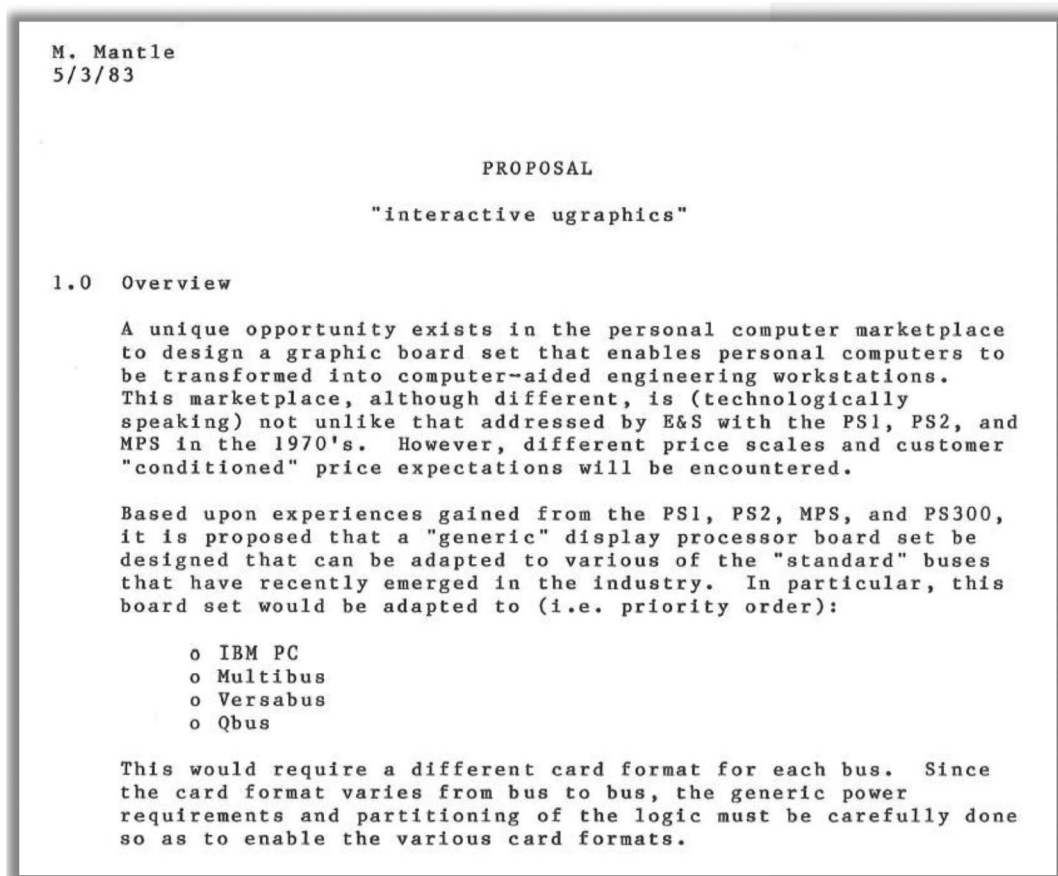


FIGURE 3. Original overview of the "Interactive ugraphics" proposal.

classes at lunch, and lots of socializing together. I still have many friendships I made with E&S employees and Picture System customers today.

But even though I had a great job and increasing responsibilities at E&S, I became disillusioned by the lack of drive by E&S management to pursue—what seemed to me—natural market growth opportunities. With a heavy heart, I listened to offers that began to come my way.

PIXAR/DISNEY, 2-D AND 3-D GRAPHICS

Ed Catmull and Alvy Ray Smith had founded the Computer Graphics Group at Lucasfilm in San Rafael, CA. In early 1986, the group had left Lucasfilm and become Pixar. I had known Ed since our computer graphics classes with Ivan and had met Alvy at the PS2 Course I taught at NYIT in 1976 after they bought a PS2 and E&S video frame buffers. They made me an offer to join Pixar not long after they left Lucasfilm. I accepted the offer and started my Pixar career.

Pixar, San Rafael, CA (1986–1991)

In May 1986, I arrived in San Rafael, CA to work at Pixar as Director of 3-D Graphics. There were two major development projects underway:

- ▶ Photorealistic 3-D Rendering software—REYES (Renders Everything You Ever Saw).
- ▶ Pixar Image Computer⁸—PIC.

I was hired to help turn these projects into deliverable products; much of the work needed to be done in just two months—before SIGGRAPH!

At SIGGRAPH 1986, Pixar was focused on delivering the following three messages.

- ▶ Pixar was now a company, owned by Steve Jobs and the employees of Pixar.
- ▶ Pixar could make stunning photo-realistic images and movies, as showcased by the *Luxo Jr.* short film, which was by far the most creative, well-crafted, and best animated and 3-D rendered short film shown at the annual SIGGRAPH Conference up until that time.
- ▶ Pixar had a hardware product to sell: the PIC, a compute and raster display system for manipulating large digital and computer generated images.

Pixar's booth was mobbed. This was the first SIGGRAPH Conference I attended for Pixar and SIGGRAPH continued to be an annual high point for me for many years.

Photorealistic 3-D Rendering—REYES

REYES was originally developed by Rob Cook, Loren Carpenter, and others before Pixar spun out from Lucasfilm. With the acclaim brought by *Luxo Jr.*, the Pixar rendering team (Pat Hanrahan and others) redoubled their effort to continue to improve the REYES rendering system.

REYES was used by Tom Porter to create two iconic images that demonstrate motion blur—the “1984 pool balls” and the Bowling Pins images (see Figure 4).

Lucasfilm's Industrial Light and Magic (ILM) was beginning to use REYES to generate special effects for movies, each of which usually had several scenes that needed special attention from the rendering team. By this time, other movie studios and special effects houses began to use REYES for creating images and special effect sequences. I began hiring programmers to enhance and support the REYES product including, Tony Apodaca, Darwin Peachy, and Dan McCoy to name just three who are still at Pixar today.

Movies, such as *The Abyss*, *Terminator*, and Disney's *Beauty and the Beast*, all had prominent scenes that took special attention from the teams. For example, *Beauty and the Beast* had a rendering issue in the ballroom scene that took many days to resolve and caused much angst at Disney. Once found (a simple computation error that occurred only under certain circumstances), the bug was relatively easy to fix. But it took a lot of effort and time to find it!

Pixar Image Computer

The PIC was a unique machine. It was general purpose in the sense that it was fully programmable for computation on images, and it was special purpose in the sense that the hardware was optimized for the image computations. PIC's computational power came from four bit-slice microprocessors that operated in parallel—one for each of four channels of 12-bit memory. The four memory channels could be displayed as R, G, and B, with the fourth channel containing % of coverage for that pixel (or Alpha). The Alpha channel also allowed multiple images in R, G, B, and Alpha format to be composited easily. The Alpha channel was a

⁸[Online]. Available: https://en.wikipedia.org/wiki/Pixar_Image_Computer

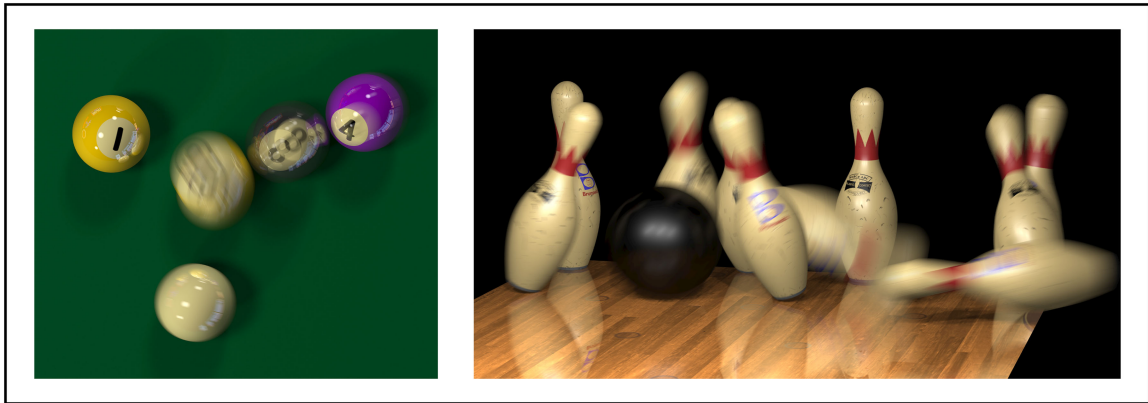


FIGURE 4. Iconic REYES images that use motion-blur.

tremendous step forward in manipulating high-quality images.^h (The technique was later popularized and introduced to graphic designers everywhere by Adobe Photoshop).

The four channels could also be used to display monochrome data. In that mode, each of the four processors would access four consecutive 12-bit pixels simultaneously.

The PIC architecture pioneered the SIMD computation mode for image processing, patents for which later helped fund Pixar when they were licensed to Intel, Motorola, Sun, AMD, and others. These companies incorporated SIMD instructions for image manipulation and other uses into their processors (e.g., for Intel, the MMX instruction set), and were their first step toward GPUs.

Pat Hanrahan wrote most of the PIC software and created the Unix man pages that were distributed digitally with the PIC. The pages formed the basis for the PIC manuals, which were packaged into the PIC Unix Software Release and manuals. The set of four manuals was packaged professionally and in time for

SIGGRAPH 1986 where the initial demos of the PIC were shown.

PIC—Philips Medical Systems (The Pegasus Project)

One of the first customers for the PIC was Philips Medical Systems who established a two-year R&D project (Pegasus) with 15 leading University Medical Centers (e.g., Duke, Columbia, Emory, Mayo Clinic, Johns Hopkins, Mallinckrodt Institute) to address various 3-D medical imaging problems. Participating sites were equipped with a PIC, a Sun workstation host, and a “tool kit” of Pixar, Island Graphics, and Cemax software to enable researchers to develop and apply 3-D imaging techniques in their research.

One of the outcomes of Pegasus was the creation of a 3-D volume rendered image from the full body CT scan of a cadaver (the Visible Human) using an algorithm developed by Pixar’s Bob Dreben (in a cooperation with Dr. Elliot Fishman of Johns Hopkins University).

I worked closely with Tom Giordano, the Philips Manager of the Pegasus, to ensure that technical problems were identified and resolved as soon as possible.

Pixar Image Computer 2 (P2)

As Pixar built sales and customer support teams for the PIC, it undertook a redesign of the product to produce a lower cost/easier-to-manufacture next generation machine that was software compatible with the earlier system.

Developing the P2 was reminiscent of my experience at E&S with the Picture System and the PS2. Most of the work was in the hardware design. Because the P2 design was compatible with the original, manufacturing, software, and documentation remained pretty much the same.

^hPhotoshop was developed in 1987 by two brothers, Thomas and John Knoll, who licensed the distribution rights to Adobe Systems Incorporated in 1988. Originally developed for the Macintosh with grayscale images on a monochrome display. John, an Industrial Light & Magic employee, recommended that Thomas turn it into a full-fledged image editing program. ILM was very familiar with the PIC and its capabilities and R, G, B, and A image formats. Photoshop essentially put a Macintosh user interface on the image processing capabilities of the PIC (which was distributed with source code for all the image processing routines), and likely provided a blue print for much of its original functionality. Photoshop, of course, provides many other capabilities but I always assumed that the PIC software was the inspiration for John Knoll’s impetus to encourage Photoshop’s move to full R, G, B, and Alpha image processing capabilities.

Philips Medical Systems PICS 2000: One of the first end-users systems that used the P2 was the Philips Medical Systems PICS 2000 (see Figure 5). Philips contracted with Pixar to produce a “turn-key” Medical Imaging System—the PICS 2000—that would accept scan data from CT scanners and enable a radiologist to view not only 2-D slices but also to process the slices to produce full 3-D rendered images. The 3-D rendering process used the algorithm developed with Johns Hopkins in the Pegasus project.

I managed the PICS 2000 project with a very small team: one programmer, manufacturing assistance to

assemble the PICS 2000 workstation and cabinet, and a contractor to oversee the FDA medical device certification required to sell these products.

I spent an inordinate amount of time managing the two Philips projects (Pegasus and PICS 2000), which resulted in a strong bond with Philips.

Electronic Light Table (ELT): TASCⁱ contracted with Pixar to produce a “turn-key” intelligence community prototype—the Electronic Light Table. The ELT, also based on the P2, would accept satellite imagery and create specially formatted images that represented the satellite data as a hierarchical set of data of increasing resolution, the 2-D analogy to 3-D level of detail. This enabled a user to navigate to a portion of the image they want to examine using lower resolution data, after which the lower resolution data were replaced by higher resolution data showing finer detail.

The ELT system used all of the hardware capabilities and high-speed data buses of the P2. It was an excellent demonstration of the power and flexibility of the P2, with performance far better than any other commercially available imaging processing system available at the time. The ELT featured 2K×2K monochrome displays (stereo viewing was optional), and a custom high-speed hard disk array—a precursor to later commercial high-speed RAID disk systems. The ELT took advantage of the P2 12-bit memory channels to display data with more levels of intensity, important for intelligence applications.

Pixar had high hopes for the ELT, and reportedly Steve Jobs expected the government to buy 2000 workstations after the delivery of the four prototypes. When this didn't happen,^j it precipitated the first Pixar course correction.

Pixar Course Correction 1988

Between 1986 and 1988, the number of Pixar employees ramped up quickly to support the PIC and animation projects. From around 40 employees when Pixar was purchased from George Lucas in 1986, it grew to 146 employees by mid-1988. The hope was that P2 sales (and its related turnkey systems PICS 2000, ELT) could lead to profitability. When it became clear that



FIGURE 5. Philips Medical Systems PICS 2000.

ⁱ[Online]. Available: https://en.wikipedia.org/wiki/TASC,_Inc.

^jThe U.S. government eventually did put out an RFP for 4000 workstations about seven years later. Glen Speckert, who spearheaded the Pixar ELT project, was an architect of the winning proposal for that RFP (“Integrated Exploitation Capability”). He disrupted again about eight years later with the Google Earth Enterprise prototype and intelligence community server roll-out.

would not be the case, drastic layoffs occurred throughout the organization.

The P2 intellectual property, inventory, and manufacturing rights, along with the P2 teams [sales, software (who just completed ELT), and manufacturing] were “sold” as a package to Vicom (note, Vicom not Viacom) for \$2M. P2 continued to be called the “Pixar Image Computer.” Eventually, the P2 teams moved to Vicom in Novato, CA from Pixar in San Rafael, CA.^k

Vicom filed for bankruptcy about six months later. They had bought the PIC, but quickly found out the market for such products was limited. Although the PIC teams supplied documents about ways to extend the P2, Vicom had no interest in evolving it. They did get a contract extension to deliver a “multispectral” version of ELT that required only software changes. But it seemed that Vicom was mostly interested in the P2 customer list.

After the layoffs and sell off of the P2 and P2 teams, the next step in the course correction was to regenerate Pixar—in a new location (Point Richmond), with new facilities, and, importantly, with new direction to produce revenue!

Move to Point Richmond

A big part of the Pixar Course Correction was to move from the buildings Pixar had leased since it was spun out of Lucasfilm (and that were next door to Lucasfilm’s Industrial Light & Magic) into a new building in Point Richmond, not far from San Rafael across the Richmond Bridge. This gave the animation production teams adequate space for the first time and also isolated the corporate, marketing, and all the other software teams into separate spaces. All shared dedicated food services and lunchroom.

A big part of this Pixar regen was an increased focus on generating revenue from sales of the REYES photorealistic rendering software and by having the animation production teams hone their creative and production skills by making 3-D animated television commercials.

Selling REYES

Pixar’s REYES photorealistic rendering software had been used by ILM and a few other special effects companies to make special effects scenes for movies. So with the expectation of driving revenue,

Pixar established a REYES sales and marketing team to establish pricing, attend trade shows, and promote the use of REYES.

Making TV Commercials

Although the animation production teams originally were concerned about making “commercials,” they warmed to the idea of contributing revenue to the company AND honing their creative and production skills so they would be ready if an opportunity to make a feature-length animated movie arose. John Lasseter, Andrew Stanton, Pete Docter, and all the Animation Production team began to make many commercials. Their early commercials for Listerine Mouth Wash, Lifesavers, and Trident chewing gum demonstrated the great creative talent that would drive Pixar’s early block-buster films (*Toy Story*, *Finding Nemo*, *Monsters, Inc.*)

Although far from cash-flow positive, the income from the commercials and the creative and animation skills that they demonstrated paved the way for Steve Jobs to continue to fund the company.

RenderMan Interface

After completing the PIC software libraries and documentation, Pat Hanrahan turned his attention to 3-D rendering and began trying to find ways to provide functionality that allowed rendering capabilities to be enhanced without changing the core code. The concept of “shaders” emerged from work by Pat and Jim Lawson.

As Pat and Jim refined their thoughts about what REYES should be, an idea crystallized that REYES could become a standardized approach to rendering 3-D images. Standardizing the basic program structure and elements of shape and structure were pretty straightforward.

The “shading language” was a foundational concept of RenderMan (the REYES successor) when it was announced. (The longer story of RenderMan is told in the 2021 article by Hanrahan and Catmull.³)

As all these elements fell into place, it was still a large leap to imagine making RenderMan a defacto standard, but Ed Catmull and Alvy Ray Smith embraced the effort. The name RenderMan came about as a nod to the ubiquitous Sony portable music player (the WalkMan) with Pixar’s hope that RenderMan would become as ubiquitous in the computer graphics community and industry.

The documentation was created in draft form by Pat Hanrahan and Jim Lawson with input from others, and the final documentation (see Figure 6) was put together by Pixar’s tech writing team.

^kSpecial thanks to Glen Speckert for providing details of the P2 team’s and ELT project’s fate after their sale to Vicom, and the eventual U.S. government RFP years later.

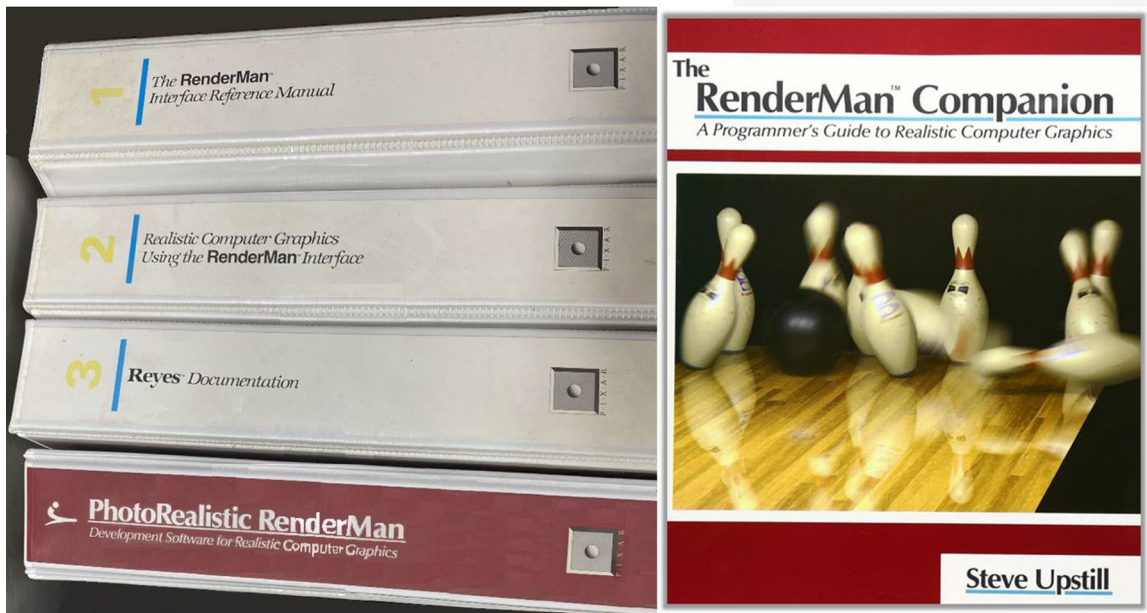


FIGURE 6. REYES, the software program, became the *Photorealistic RenderMan* product with the *RenderMan Companion* book.

The final piece of documentation that helped establish RenderMan become a commercial success was *The RenderMan Companion* book, written by Steve Upstill and published by Addison-Wesley (see Figure 6) that detailed how to effectively use RenderMan. I managed the interactions with the Addison-Wesley Editor needed to get the book published.

Alvy spent a bit of time coming up with a logo that he felt captured the essence of RenderMan.



Pixar also made a strong move toward making RenderMan a defacto industry standard by creating the RenderMan Advisory Board, which I chaired for several years. Pixar used the Advisory Board, composed of a number of key RenderMan users and potential users, as a sounding board for changes that were being considered to the RenderMan Interface as well as providing a formal conduit for suggestions and improvements that Advisory Board members felt were needed.

Pixar Course Correction 1990

Although there was revenue coming in from sales of RenderMan software as well as for the many commercials, the animation production team was cranking

out, Pixar was still nowhere near cash-flow positive. The company had done many amazing things and had a strong reputation and image in the graphics industry. But it was still burning through too much cash—and Steve Jobs was getting tired of funding it.

To reduce the burn rate, Steve declared that another layoff had to occur and that major costs had to be reduced. Steve and Ed arrived at a plan to downsize Pixar by about 30 employees, including some of the highest-paid employees: the President, CFO, VP Marketing, VP Engineering, Purchasing, and other support and nonessential staff.

Ed and Alvy's roles were not directly affected, although Ed assumed the joint role of President/CTO.

I was assigned the unenviable task of terminating the President, CFO, plus several others, and afterward was promoted to General Manager of Graphics Development.

Other RenderMan Products

In a quest to capitalize on the attention that RenderMan was gaining, the smaller Pixar began two new product initiatives:

- 1) An interactive version of RenderMan.

QuickRenderMan—a “lightweight” implementation of RenderMan designed to be a contender for a 2-D/3-D interactive graphics standard. Again Pat Hanrahan led the charge. We engaged with Microsoft in

an attempt to do with graphics what Adobe had done with Postscript for page description and printing. However, dealing with Microsoft was difficult (at best) and, in the end, impossible. They had learned their lesson with Adobe's Postscript and were not about to make that kind of mistake again. Without the prospect of partnering with Microsoft, work on QuickRenderMan was discontinued.

2) RenderMan consumer products.

MacRenderMan, *Typestry*, and *Showplace* were "shrink-wrapped" consumer software that enabled 3-D objects to be, respectively: rendered, designed from text fonts, and created from 3-D objects. The plan was to sell these products in software stores or using other distribution channels. However, the market was limited.

PCRenderMan—this implementation for the PC/DOS, with the full capabilities of RenderMan available—was only possible by using Phar Lap DOS extender that allowed access to memory beyond the 640 KB limit of DOS.

Although these consumer products were good and the packaging professional with excellent shelf presence, the limited processing power of Macs and PCs made the rendering quite slow and not very many units were sold. This was long before MS Windows, and MS DOS discouraged much interest in *PCRenderMan*.

Another Pixar Project: *ICEMan*

As RenderMan acceptance as a defacto standard accelerated, Alvy embarked on developing a sophisticated approach to image computing, called *ICEMan* (Image Computing Environment). *ICEMan* provided a procedural approach for defining image computing operators and operations. His primary collaborator was programmer Bruce Perins, later a prominent figure in promoting the licensing and use of Open Source Software.

Alvy and Bruce worked hard on *ICEMan* for many months, with several internal releases for feedback and evaluation. But before the project was completed, Alvy and Steve Jobs had a dispute that resulted in Alvy leaving Pixar to found his own software company, Altamira Software. Eric Lyons and Nick Clay co-founded Altamira.

Altamira produced a PC-based product, *Altamira Composer*, which pioneered object-based drawing and image editing. *Composer*, based upon *ICEMan* principles, was one of the first to bring important tools, such as antialiasing and image sprites to the desktop. Altamira was later bought by Microsoft and published

a series of Microsoft products based upon *Composer*—*Picture It!* and *PhotoDraw*. Alvy joined Microsoft and became their first Graphics Fellow.

Pixar and Disney (1990–Today)

While the animation teams were honing their creative and production skills making commercials, it was no secret that Ed, Alvy, and all of Pixar wanted to make a feature length computer generated animated movie.

Since Ivan's CG class at the U of U in 1971, Ed had shown dogged determination to invent the technology and recruit the talent to make that possible. He kept this goal in front of him and his teams first at NYIT, then at Lucasfilm, and now, especially, at Pixar. An outrageous goal, such as making the first full length animated 3-D computer generated film was exactly the kind of thing that would fire Steve Jobs up and keep him investing in Pixar during the years when it was losing money.

To convince Steve that such an idea was feasible, John Lasseter, Andrew Stanton, Pete Docter, and others began working on a storyboard for a film. Disney was the right partner for Pixar, so Steve, Ed, and John embarked on a campaign to get Disney to fund ("green light") the movie.

Disney "Greenlights" *Toy Story*

After a lot of negotiations that strained the relationship between Steve Jobs and Disney's CEO Michael Eisner, an agreement was finally reached where Disney agreed to fund and distribute Pixar's first movie with options for follow-on movies if the first one was successful. This was announced in early July, 1991.

The rest of Pixar's story is the stuff of Hollywood legend. No movie studio had ever had such a record of consecutive box office hits that would equal Pixar's. In turn, that success would lead to Steve Jobs selling Pixar to the Walt Disney Company in 2006 for over \$7 billion. Jobs became Disney's largest individual shareholder with a 7% stake.

Leaving Pixar Animation Studios

When it became clear that Pixar was turning into an animation studio, I began to question what my role would be. I also did not have confidence that Pixar would ever be a public company—no pure animation company had ever gone public (although Pixar did, as they released *Toy Story*, their first feature-length film).

I did not foresee the immense technical challenges that Pixar would need to overcome to keep pushing the state of the art in their animated films. There probably would have been plenty for me to do.

But I felt it had come time for me to move on, so I began to listen to siren songs from some successful “PC software” companies. We had created several RenderMan products for the Mac and PC and I had been following the technology and growing popularity of personal computers and consumer software closely.

In late 1991, Brøderbund Software made me a compelling offer—VP of Engineering/CTO overseeing all of their software development, graphics design/animation, and sound/music departments. Quality assurance and information technology would also report into me.

When I gave my notice at Pixar, they were surprised. Few people ever leave Pixar voluntarily. Steve Jobs tried to talk me out of leaving. I remember sitting in my office with Steve as he made some promises I doubted he could (or would) fulfill. As he left, I remember thinking—“this reality distortion field is so real; if he would have told me my white wall was blue, I probably would have believed him!”

But Brøderbund Software (publisher of multimedia products and games, such as Print Shop, Carmen San-diego, Kid Pix, Myst, Living Books, and many others) had made me a great offer, was poised to go public, and I would have a title and span of control I could never hope to obtain at Pixar. So, despite Steve’s reality distortion field, I left Pixar in late 1991 to join Brøderbund.

CONCLUSION

Working at E&S and Pixar were dreams come true for me.

At E&S, I met and interacted with key technical people at many of the top aerospace and automotive companies, plus many researchers who were seeking advice on how best to use their Picture System products. Molecular modeling was a favorite application for E&S graphics systems, and I had the privilege to get to know some of those researchers well.

My experience at Pixar was similar, but with different hardware/software systems and customers. I met and interacted with key people at some of the top medical researcher centers by way of the Philips Pegasus project, and many of the top special effects groups at movie studios and effects houses that were using REYES, and later RenderMan.

Doing this, all the while overseeing major technical initiatives, people, and teams.

However, my continual desire to learn new things and have new experiences led me to spend more years working in other domains than I did building computer graphics systems and software:

- › Computer-aid design (total of three years).
- › Web-based services and audio/video content and technologies (12 years).
- › Consumer software products (total of 18 years).

Computer graphics has been, and always will be, a big part of my life. It was a key ingredient in all of the companies I worked for. I am blessed to have loved what I was paid to do—learn about new technologies and products, oversee interesting projects, and work with many smart and interesting people.

I’ve learned many things over the 50+ years of my career. Looking back, perhaps the most important part of my job was not the technology; it was the people I met, hired, worked with, managed, and often befriended at E&S, Pixar, and all the companies I worked at.

“One More Thing”

Starting around 2001, I began a decade long personal project: coauthoring a book (with my friend and fellow software development manager, Ron Lichty) that distilled many of the lessons we had learned managing software people and teams. We took it to Addison-Wesley and Peter Gordon, the Editor I worked with at Pixar on the *RenderMan Companion*, who encouraged

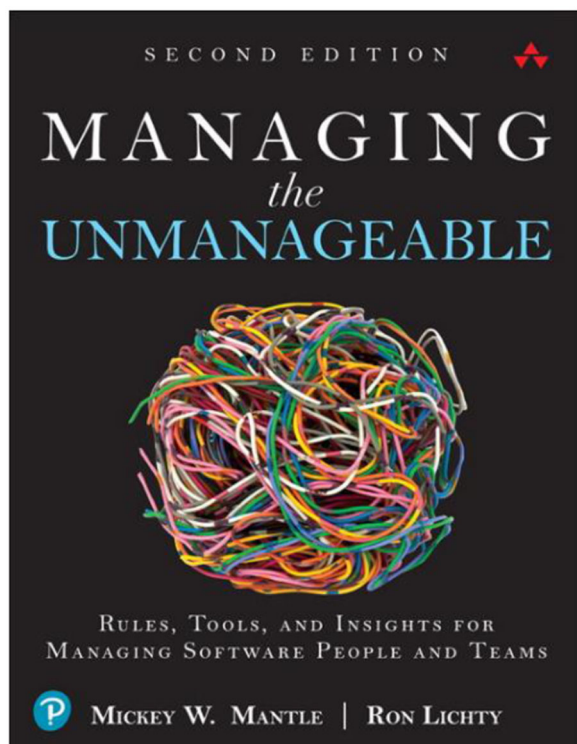


FIGURE 7. *Managing the Unmanageable* (second edition).

us to write the book. Over a decade later, that book was published and is now in its second edition (see Figure 7).

Managing the Unmanageable can be read sequentially, Chapters 1–10, or used as a reference for anyone who manages others, especially software people and teams. It is accompanied by a website^l that contains presentations, interviews, additional rules-of-thumb, and much more, which augments and amplifies the information contained in the book.

Today I am retired, but I do consulting, mostly with startups. My current technical interests focus on software development metrics and co-host a quarterly Engineering Productivity Tools Meetup with invited presenters and discussion^m; I'm still trying to find the best ways to effectively manage programmers and make great products.

ACKNOWLEDGMENTS

Special thanks to Dave Kasik for encouraging me to write this article, and helping me to constrain the story I originally set out to tell. Many thanks to Dave, Mary Whitton, and Chris Johnson, whose many suggestions and edits made this article much better.

REFERENCES

1. *The Big 50: Celebrating 50 Years of SIGGRAPH Conferences*, IEEE Comput. Graphics Appl., Aug./Sep. 2023.
2. I. E. Sutherland, "Polygon sorting by subdivision: A solution to the hidden-surface problem," Oct. 1973.
3. P. Hanrahan and E. Catmull, "The design of RenderMan," IEEE Comput. Graphics Appl., vol. 41, no. 4, pp. 7–15, Jul./Aug. 2021, doi: [10.1109/MCG.2021.30853741](https://doi.org/10.1109/MCG.2021.30853741).

MICKEY W. MANTLE is with Mantle Consulting Napa Valley, CA, 94558, USA. Contact him at mickey@mmantle.com.

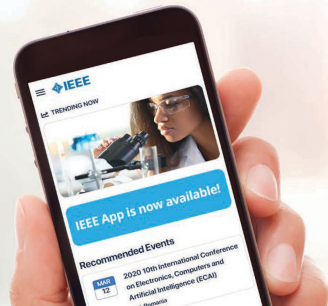
Contact department editor Dave Kasik at dave.kasik@gmail.com, department editor Chris Johnson at crj@sci.utah.edu, or department editor Mary C. Whitton at mcwhitton@gmail.com.

^l[Online]. Available: <https://managingtheunmanageable.net>

^m[Online]. Available: <https://meetup.com/eng-productivity-tools/>

THE IEEE APP:

Let's stay connected...



Stay connected by discovering the valuable tools and resources of IEEE:

-  Create a personalized experience
-  Get geo and interest-based recommendations
-  Schedule, manage, or join meetups virtually
-  Read and download your IEEE magazines
-  Stay up-to-date with the latest news
-  Locate IEEE members by location, interests, and affiliations



Download Today!

